Improving Efficiency in Neural Network Accelerator using Operands Hamming Distance Optimization

Meng Li*, YiLei Li*, **Pierce Chuang**, Liangzhen Lai, and Vikas Chandra

EMC2 Workshop @ NeurIPS 2019

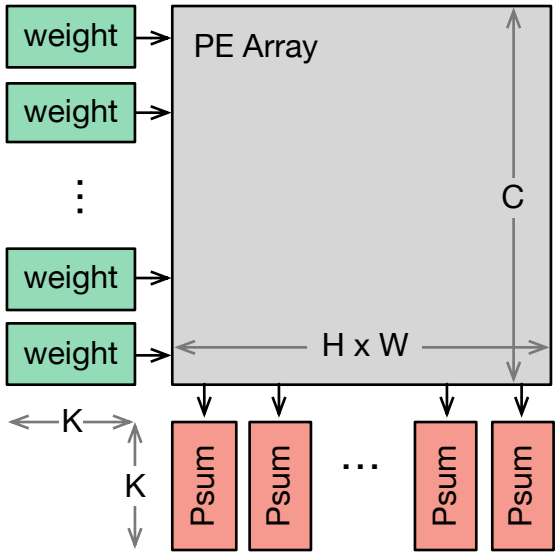
Facebook Silicon AI Research

Motivation

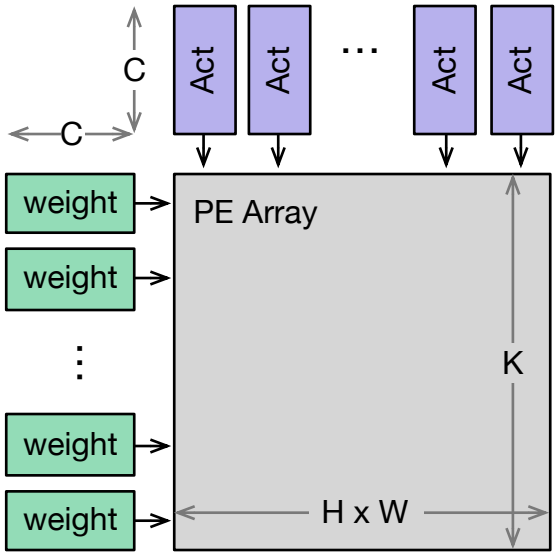
Dataflow processing is widely exploited to amortize memory access energy

Datapath energy becomes important for dataflow accelerators

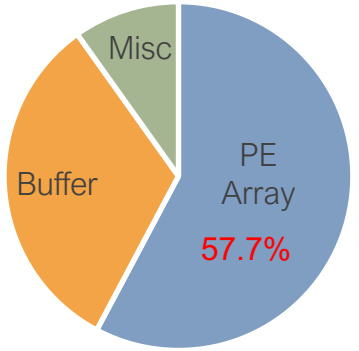
- Consist of compute energy in process elements (PEs) and data propagation energy among PEs



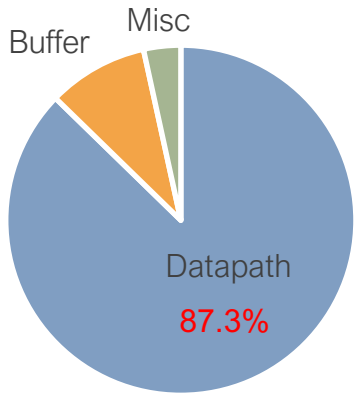
Input Stationary



Output Stationary



Thinker [Yin+, JSSC'18]



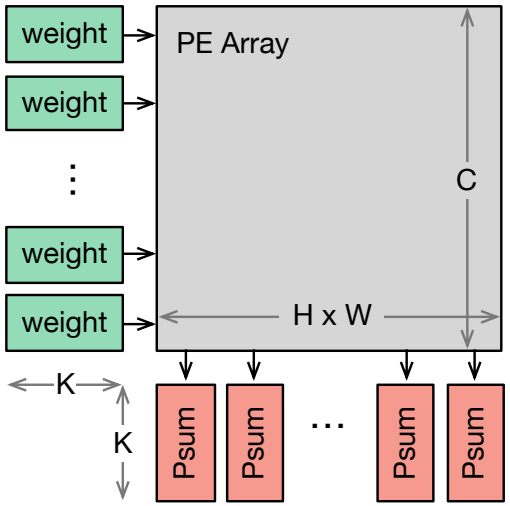
ShiDianNao [Du+, ISCA'15]

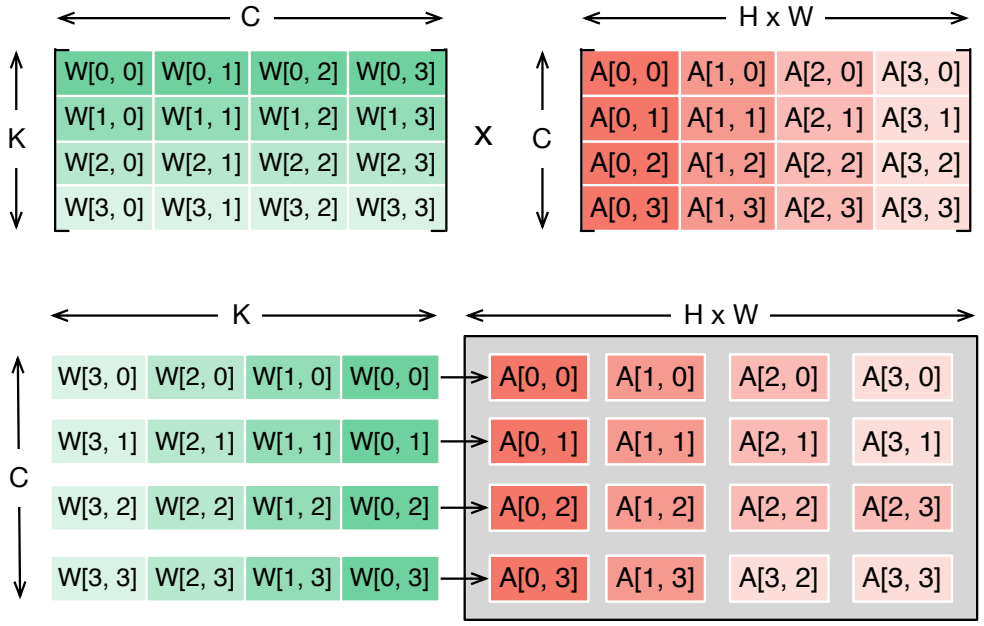
Motivation

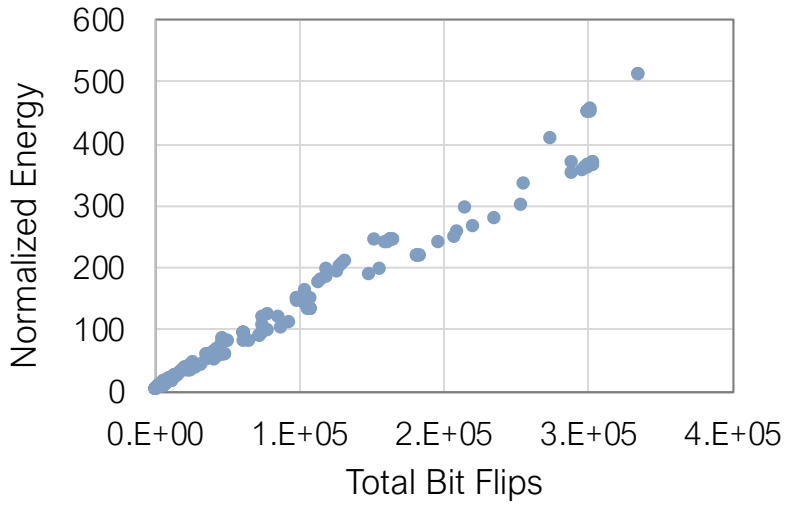
In dataflow processing, operands are streamed into the compute array

Datapath energy is determined by the total bit flips induced by operand streaming

Target: propose post-training and training-aware techniques to reduce bit flips of weight streaming





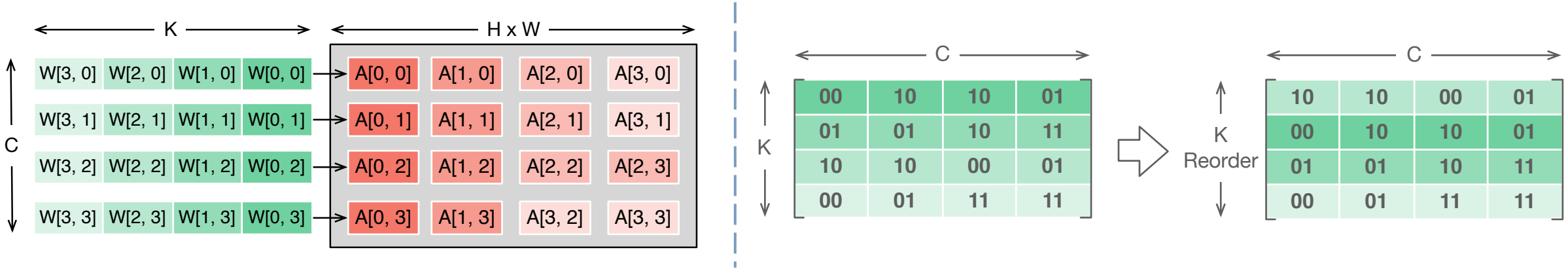


K, C, H, W denotes output channel, input channel, output height, and output width, respectively

Post-Training Optimization: Output Channel Reordering

To reduce bit flips, the most straight-forward technique is output channel reordering

- Output channel reordering can be mapped to a traveling salesman problem, which can be approximately solved with efficient greedy algorithms



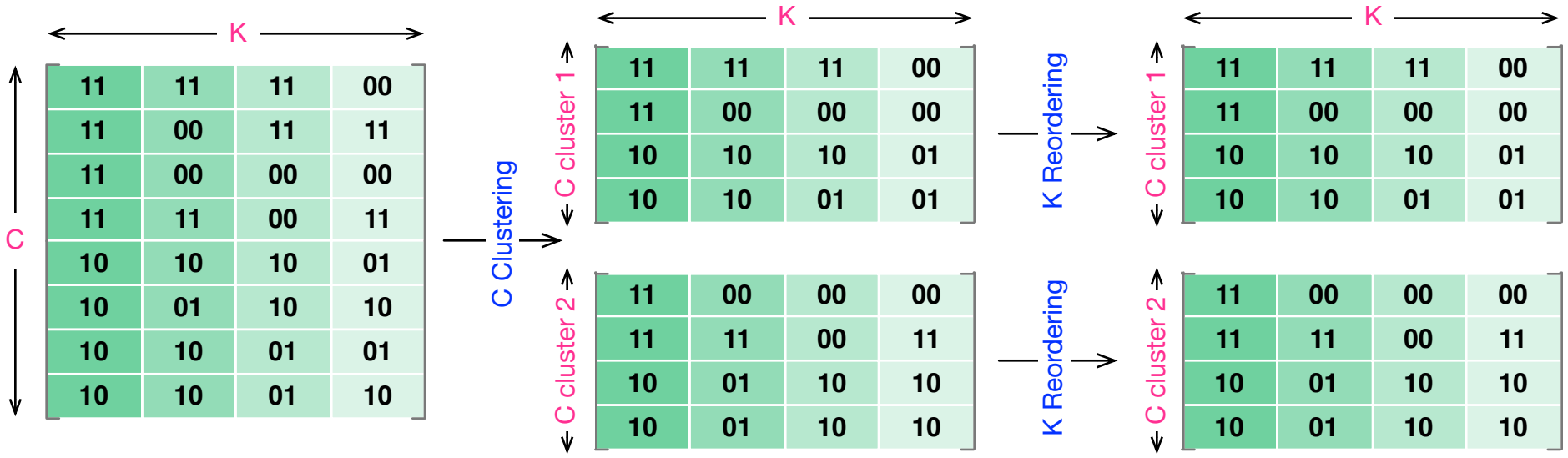
Post-Training Optimization: Input Channel Clustering

For most networks, the channel dimension can be larger than the compute array size

Weight matrices need to be segmented first and then fed into compute array

- Each weight sub-matrix can use different output channel orders
- Before segmenting the weight matrix, different input channels can be clustered first

Propose an iterative assignment and update approach for input clustering



Experimental Results

Post-training optimization technique comparison

- Use 1x1 Conv in MobileNetV2 and 3x3 Conv in ResNet26 for evaluation

Combine post-training and training-aware optimization

- Incorporate bit flip loss into the loss function
- Use MobileNetV2 trained on Cifar100 for evaluation